Conditional Graph Diffusion with Topological Constraints for Brain Network Generation

Joonhyuk Park[1], Donghyun Lee[1], Guorong Wu[2], and Won Hwa Kim[1]

[1] Pohang University of Science and Technology, Pohang, South Korea
{pjh1023, wonhwa}@postech.ac.kr
[2] University of North Carolina at Chapel Hill, Chapel Hill, USA

Abstract. The acquisition of structural brain network data is inherently challenging due to high costs of Diffusion Tensor Imaging (DTI) and the complexity of data processing such as tractography. Moreover, medical datasets often exhibit severe class imbalance where the sample size of healthy subjects highly exceeds that of diseased. While recent graph generation models offer a potential solution, its application to brain networks is understudied as they often underestimate preserving topological feature which is an essential biomarker. To address these limitations, we propose a conditional graph diffusion model that ensures high-fidelity graph generation by leveraging persistent homology. Specifically, we introduce a Conditional Graph Diffusion (ConGD) method that utilizes Condition Infused Attention (CIA) module with class and structure conditioning, to enable the targeted synthesis of brain networks, and Topology Aligning (TA) regularization to enforce topological consistency. Experiments on the Alzheimer's Disease Neuroimaging Initiative (ADNI) dataset demonstrate that our approach provides high-fidelity synthetic brain networks under label conditions, which are further validated for improving predictive performance through downstream graph classification tasks.

Keywords: Conditional Graph Diffusion · Persistent Homology · Brain Network.

1 Introduction

Brain network analysis via the lens of graph representation has been a mainstream approach for understanding interactions betweeen different brain region of interest (ROI) and their relationship to neurological disorders and cognitive function [4, **?**]. However, curating the brain network dataset in the initial stage faces significant challenges, as constructing brain networks typically requires expensive neuroimaging scans as well as preprocessing pipelines, e.g., tractography on Diffusion Tensor Imaging (DTI) [24]. Moreover, neuroimaging datasets often exhibit severe class imbalance, where healthy subjects far outnumber the diseased. While graph generative method is a potential solution, its application to brain networks remains limited due to their complex structure.

Recent Graph generation methods aim to synthesize graphs that resemble the distribution of real graphs. Existing methods, such as graph recurrent neural networks [29], variational autoencoders [22], and diffusion-based models [13, 26], have demonstrated promising results in generating graphs with high resemblance. However, these approaches primarily focus on preserving statistical properties, such as degree distributions and clustering coefficients, often neglecting preserving global topological structures. Such approaches are particularly problematic for brain networks, where topological features serve as critical biomarkers for neurological conditions [20, 23]. For example, deploying graph diffusion model directly for brain network generation will synthesize edges inter-connecting left and right hemispheres, which highly deviates from the true data. Therefore, generating synthetic brain networks under topological and disease conditions is crucial for augmenting limited data to aid downstream analysis.

To address these challenges, we propose a novel Conditional Graph Diffusion (ConGD) method. Unlike conventional graph generative models, ConGD generates graphs under several conditions. There are two key components: 1) Condition Infused Attention (CIA) module that leverages robust embeddings of conditions as a bias term in a transformer architecture to enhance the denoising network's ability to provide representations consistent on the conditions, and 2) Topology Alignment (TA) based on persistent homology, which enforces consistency between the homological features of real and generated graphs via a 1-Wasserstein distance. Also, ConGD employs a class-conditioned diffusion to enable class-targeted augmentation, to address the class imbalance problem.

Contributions. Our main contributions are as follows: **1)** we propose a novel graph diffusion model that preserves topological fidelity, **2)** we design a CIA module, which improves structural consistency in generating disease-specific brain networks and alleviates class imbalance problem, **3)** we introduce the topology alignment that enforces homological consistency into the generation process. Through comprehensive experiments on downstream graph classification, we validate the brain networks generated by ConGD on the Alzheimer's Disease Neuroimaging Initiative (ADNI) dataset, demonstrating superior performance.

2 Related Work

2.1 Discrete Diffusion Models for Graph Generation.

Recently, diffusion-based models [2, 13, 26] have emerged as a promising graph generation approach. In particular, discrete diffusion methods proposed in [1] and [26] have demonstrated strong performance in capturing the joint distribution of nodes and edges. Diffusion models consist of two components: the forward and reverse process, each conducted by a noise model and a denoising network.

Forward process. Let $G = (V, E)$ be a graph, where V is the set of N nodes, and $E \subseteq V \times V$ is the set of edges. Treating the node and edge attributes as categorical distributions with cardinalities d_V and d_E, the node and edge attribute matrices are given as $X \in \mathbb{R}^{N \times d_V}$ and $A \in \mathbb{R}^{N \times N \times d_E}$, respectively. In

the discrete diffusion setting, a noise model in the forward process progressively corrupts a clean initial graph $G^0 = (X^0, A^0) = G$ to a noisy graph $G^t = (X^t, A^t)$, for time step $t = 1, 2, ..., T$. Specifically, the noise is represented as transition matrices $Q_X^t \in \mathbb{R}^{d_V \times d_V}$ and $Q_A^t \in \mathbb{R}^{d_E \times d_E}$, such as a uniform distribution, which shifts the distribution of X^{t-1} and A^{t-1} to that of X^t and A^t, imposing noise at each time step [1]. By multiplying the transition matrix to X^0 and A^0 for t times, we can obtain a noisy graph G^t.
Reverse process. The denoising network ϕ inverts the forward process by estimating the categorical probability vector $\hat{p}_{G^0}$ from G^t. During inference, a graph is synthesized by iteratively estimating $\hat{p}_{G^0}$ from G^t with ϕ and imposing noise back to the estimate $\hat{p}_{G^0}$ to obtain G^{t-1}, progressing from $t = T$ to 1.

2.2 Persistent Homology on Graphs

Persistent homology (PH) is a method used to encode global structural properties of data [5, 9]. It captures topological features, e.g., connected components or loops, by tracking their *birth* and *death* across a *filtration*, i.e., a nested sequence of subgraphs. By encoding the entire filtration, PH provides a rich representation of global structure that may not be readily apparent in the original graph.

Persistent diagram is a primary representation method of PH. Given a filtration, suppose a topological feature is born at i-th subgraph and died at j-th subgraph, i.e., that it persists from i to j. This feature can be represented as a *persistence barcode*, a tuple (i, j) of birth and death pair. Plotting these barcodes as points on $\mathbb{R}^2$ plane, one can obtain a persistence diagram $\mathcal{D}_G$ of a graph G:

$$\mathcal{D}_G = \{(b, d) \mid \text{persistence barcode of } G\} \subseteq \mathbb{R}^2. \tag{1}$$

Persistence landscape [3] is a vectorization method that transforms the topological features in a persistence diagram $\mathcal{D}_G$ into a fixed-size vector representation μ_G, by encoding them as a sequence of piecewise linear functions. These representations, the persistence diagram $\mathcal{D}_G$ and the topological feature vector μ_G, encode complete persistent homology information of a given graph [3, 9].

Persistent homology has been increasingly utilized to inject topological information into learning across various domains, including medical images [7, 19], point clouds [17], 3D shapes [10], and graph representation learning [12, 28]. Additionally, topology generative models have also been proposed in domains such as images and molecular structures [27, 30]. However, these approaches remain inapplicable to discrete graphs with sparse and irregular connectivity, limitations that are particularly critical in brain networks where global topology serves as a crucial biomarker for disease characterization [20, 23]. Besides PH, several studies have utilized topology of brain networks, derived from topological data analysis, to classification tasks, achieving meaningful performance gains [11, 18].

3 Graph Diffusion with Conditions

In this section, we propose ConGD, a Conditional Graph Diffusion framework for class-conditioned graph generation with high structural fidelity. Following

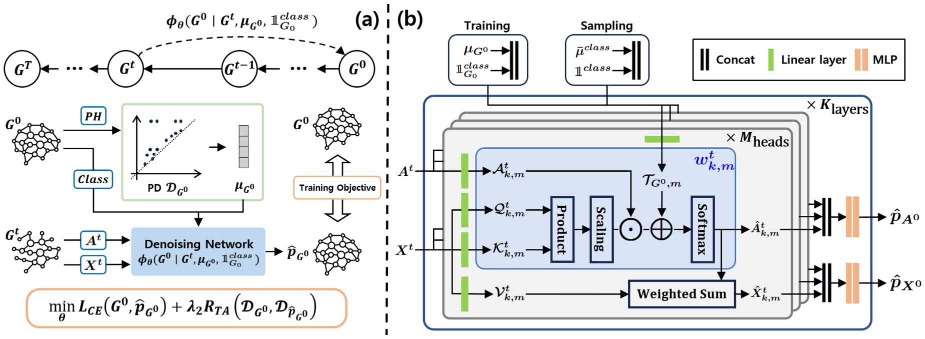


Fig. 1: The overall framework of ConGD: (a) the training process and (b) the denoising network ϕ_θ with K transformer layers with M multi-headed CIA modules.

[1, 26], the diffusion process of ConGD is defined on a discrete space, with the forward and reverse process acting on the node and edge attributes. Specifically, we extend the graph transformer network [8] as the denoising network ϕ_θ, parametrized by θ, to estimate the probability vector $\hat{p}$ of the nodes and edges of G^0, i.e., $\hat{p}_{G^0} = (\hat{p}_{X^0}, \hat{p}_{A^0})$. An overview of our framework is illustrated in Fig. 1.

3.1 Class-conditioned Graph Generation

Condition Infused Attention Module. An important extension of the graph transformer in ConGD is the Condition Infused Attention (CIA) module, which enhances class-aware representation learning by integrating class-conditioned topological embeddings. In addition to the noisy graph G^t, which is primarily utilized in graph transformer models [8, 21], CIA utilizes the topological feature vector μ_{G^0} obtained via persistence landscape (Sec. 2.2) and the class label one-hot vector $\mathbb{1}^{class}_{G^0}$.

In the denoising network $\phi_\theta(G^0|G^t, \mu_{G_0}, \mathbb{1}^{class}_{G^0})$, CIA is repeated for K transformer layers, where each layer consists of M multi-headed CIA modules. At each k-th layer, CIA operates on the node and edge attribute, X^t_k and A^t_k, which are initialized from the noisy graph (X^t, A^t). Similar to standard graph transformers, the query $\mathcal{Q}^t_k$, key $\mathcal{K}^t_k$, and value $\mathcal{V}^t_k$ matrices are obtained by applying distinct linear transformations to X^t_k, while the edge embedding $\mathcal{A}^t_k$ is derived from A^t_k. Additionally, we define a class-conditioned topological embedding, $\mathcal{T}_{G^0} = W_\mathcal{T}\left[\mu_{G^0} \| \mathbb{1}^{class}_{G^0}\right]$, where $W_\mathcal{T}$ is a learnable weight matrix and $\|$ denotes concatenation. Using these representations, the attention score $w^t_{k,m}$ and its intermediate score $\alpha^t_{k,m}$ in the m-th head of CIA is computed as:

$$\alpha^t_{k,m} = \frac{\mathcal{Q}^t_{k,m} \cdot {\mathcal{K}^t_{k,m}}^T}{\sqrt{d}}, \qquad [w^t_{k,m}]_{i,j} = \frac{\exp\left(\left[\alpha^t_{k,m} \cdot \mathcal{A}^t_{k,m} + \mathcal{T}_{G^0,m}\right]_{i,j}\right)}{\sum_j \exp\left(\left[\alpha^t_{k,m} \cdot \mathcal{A}^t_{k,m} + \mathcal{T}_{G^0,m}\right]_{i,j}\right)}, \tag{2}$$

where d denotes the dimension of $\mathcal{Q}^t_{k,m}$, and $[\cdot]_{i,j}$ indicates the (i,j)-th element of the matrix.

Using the $w_{k,m}^t$, intermediate node and edge embeddings, $\hat{X}_k^t$ and $\hat{A}_k^t$, are derived by aggregating the outputs of the M multi-headed CIA as:

$$\hat{X}_k^t = [w_{k,1}^t \mathcal{V}_{k,1}^t \| \cdots \| w_{k,M}^t \mathcal{V}_{k,M}^t], \qquad \hat{A}_k^t = [w_{k,1}^t \| \cdots \| w_{k,M}^t]. \tag{3}$$

The transformer layer then updates the node and edge attributes by projecting the aggregated embeddings to their original dimensions via an MLP:

$$X_{k+1}^t = W_{X,2}\,\mathrm{ReLU}\left(W_{X,1}\hat{X}_k^t\right), \quad A_{k+1}^t = W_{A,2}\,\mathrm{ReLU}\left(W_{A,1}\hat{A}_k^t\right), \tag{4}$$

where $W_{\cdot,1}$ and $W_{\cdot,2}$ are the learnable weight matrices for a 2-layer MLP. The final node and edge attributes, obtained through K iterative transformer layers, are used to estimate the categorical probability vector $\hat{p}_{G^0}$ of graph G^0 under topological and class-wise conditions.

Note that CIA utilizes class-conditioned topological embedding $\mathcal{T}_{G^0}$ as a global attention bias term to guide the attention to capture both class information and topological features within the graph. Unlike conventional graph transformers, which integrate edge embeddings into node-wise attention [8], capturing local structures, CIA leverages $\mathcal{T}_{G^0}$ to encode global structural properties from the topological feature vector μ_{G^0} to the attention computation. By leveraging $\mathcal{T}_{G^0}$, CIA enhances the estimation of topology-aware node and edge embeddings, refining the estimation of the topology-aware probability vector $\hat{p}_{G^0}$.

Class-conditioned Topological Embedding for Sampling. Class conditioning in $\mathcal{T}_{G^0}$ plays a more prominent role in sampling. During training, CIA utilizes μ_{G^0} computed from the original graph via persistence landscape, along with $\mathbb{1}_{G^0}^{class}$. However, in the reverse diffusion process, the original graph is unavailable, preventing direct computation of μ_{G^0}. To address this issue, we replace μ_{G^0} with the class-wise averaged persistence landscape $\bar{\mu}^{class}$ precomputed from the training dataset, and change the class label vector $\mathbb{1}_{G^0}^{class}$ with the desired class label vectors $\mathbb{1}^{class}$. Denoting the transition matrix of node and edge attributes as Q_X^t and Q_A^t (from Sec. 2.1), the sampling process is done as:

$$(\hat{p}_{X^0}, \hat{p}_{A^0}) = \phi_\theta(X^0, A^0 \mid X^t, A^t, \bar{\mu}^{\mathrm{class}}, \mathbb{1}^{class}), \tag{5}$$

$$X^{t-1} = \hat{p}_{X^0} \prod_{s=1}^{t-1} Q_X^s, \quad A^{t-1} = \hat{p}_{A^0} \prod_{s=1}^{t-1} Q_A^s. \tag{6}$$

By repeating Eq. (5) and (6) for $t = T, T-1, ..., 1$, we can generate a graph conditioned on a desired class through sampling from the final categorical distribution $\hat{p}_{X^0}$ and $\hat{p}_{A^0}$. The substitution of conditional embedding $\mathcal{T}_{G^0}$ allows the topology-aware attention mechanism to retain class-specific structural information during sampling, ensuring that the generated graphs align with the distribution of the target class in the training dataset. The effectiveness of class-conditioned topological embeddings in improving graph generation is further demonstrated in Sec. 4.

3.2 Objective Function

The final objective of ConGD is to produce accurate estimation of the probability vector $\hat{p}_{G^0}$, which aligns the joint distribution of nodes and edges while preserving conditional properties, i.e., disease label and topological structure. We first aim to ensure that the probability vectors produced by ϕ_θ conditioned on the label as well as the structure closely approximate the ground truth attributes of the clean graph G^0. This is achieved by minimizing the cross entropy loss over all nodes and edges:

$$\begin{aligned} L_{\mathrm{CE}}\left(G^0, \hat{p}_{G^0}\right) &= L_{\mathrm{CE}}\left(X^0, \hat{p}_{X^0}\right) + \lambda_1 L_{\mathrm{CE}}\left(A^0, \hat{p}_{A^0}\right) \\ &= -\sum_{i=1}^{N} x_i \log\left(\hat{p}_{x_i}\right) - \lambda_1 \sum_{i=1}^{N}\sum_{j=1}^{N} a_{i,j} \log\left(\hat{p}_{a_{i,j}}\right), \end{aligned} \tag{7}$$

where $\lambda_1 \in (0, 1]$, x_i and $a_{i,j}$ are the i-th node and its edge connected to node j.

To further ensure that the generated graphs accurately reflect the topological characteristics of the original graphs, we aim to minimize the discrepancy between the conditionally generated and the real graphs via persistence diagrams. For a given original graph G^0 and the estimated $\hat{p}_{G^0}$, their persistence diagrams $\mathcal{D}_{G^0}$ and $\mathcal{D}_{\hat{p}_{G^0}}$ are derived as in Eq. (1). Considering the diagrams as distributions [16], we measure the discrepancy between the two distributions via 1-Wasserstein distance as:

$$R_{\mathrm{TA}}(\mathcal{D}_{G^0}, \mathcal{D}_{\hat{p}_{G^0}}) = \sum_{\mathbf{z}\in\mathcal{D}_{G^0}} \left([\mathbf{z}_{\mathrm{birth}} - \gamma^*(\mathbf{z})_{\mathrm{birth}}]^2 + [\mathbf{z}_{\mathrm{death}} - \gamma^*(\mathbf{z})_{\mathrm{death}}]^2\right), \tag{8}$$

where γ^* is the optimal bijection between $\mathcal{D}_{G^0}$ and $\mathcal{D}_{\hat{p}_{G^0}}$. As one-to-one bijection is infeasible, since the number of elements in the $\mathcal{D}_{G^0}$ and $\mathcal{D}_{\hat{p}_{G^0}}$ are different, we padded the diagrams with dummy elements, i.e., points with identical birth and death pair that carry minimal topological information, to ensure the bijection to exist [14].

Finally, the training objective of ConGD is formulated as a combination of (7) and (8) as:

$$\theta^* = \arg\min_\theta L_{\mathrm{CE}}\left(G^0, \hat{p}_{G^0}\right) + \lambda_2 R_{\mathrm{TA}}(\mathcal{D}_{G^0}, \mathcal{D}_{\hat{p}_{G^0}}), \tag{9}$$

where $\lambda_2 \in (0, 1]$ controls the effect of regularization.

4 Experiments

4.1 Experimental details

Dataset. We validate our method using structural brain connectivity data derived from Diffusion Weighted Imaging (DWI) in the Alzheimer's Disease Neuroimaging Initiative (ADNI), processed with an in-house tractography pipeline. Using the Destrieux atlas [6], each brain network is represented as an undirected

Table 1: Quantitative comparison on downstream graph classification task. The best scores are highlighted in **Bold**.

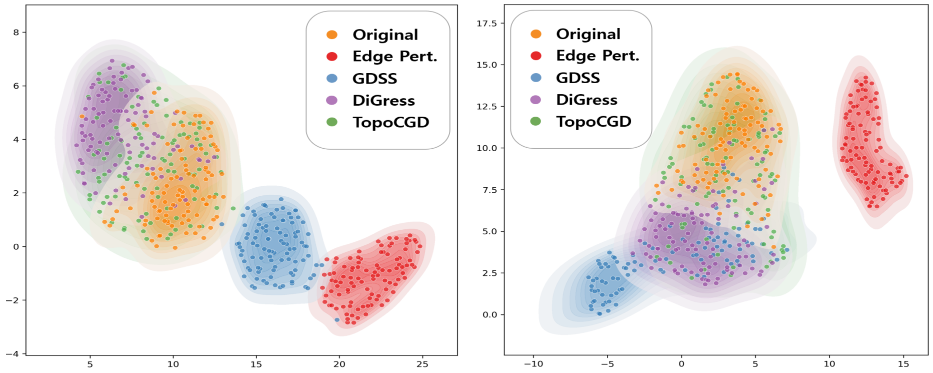
Method	Conventional Methods								Graph Methods							
	SVM				MLP				GCN [15]				GAT [25]			
	Acc.	Prec.	Rec.	f1.	Acc.	Prec.	Rec.	f1.	Acc.	Prec.	Rec.	f1.	Acc.	Prec.	Rec.	f1.
Binary Classification (CN/AD)																
No Aug.	66.9	61.1	62.4	62.4	72.3	70.1	64.4	63.3	72.6	58.1	56.5	53.8	69.1	58.3	60.4	58.0
Edge Pert.	68.2	68.5	68.2	68.0	72.5	74.7	72.5	71.5	70.5	70.6	70.5	67.8	66.5	68.1	66.5	62.7
GDSS [13]	67.8	68.5	67.8	67.4	68.8	71.3	68.8	67.9	69.2	71.3	69.2	67.2	70.6	71.7	70.6	70.2
DiGress [26]	66.9	67.1	66.9	66.7	70.4	70.9	70.4	70.2	70.9	71.4	70.9	70.7	68.4	63.6	68.4	64.8
ConGD	**71.3**	**71.6**	**71.3**	**71.2**	**77.3**	**78.7**	**77.3**	**77.0**	**78.0**	**78.4**	**78.0**	**77.8**	**71.3**	**72.6**	**71.3**	**70.8**
5-way Classification (CN/SMC/EMCI/LMCI/AD)																
No Aug.	**44.2**	40.8	36.4	36.3	**41.6**	27.7	27.6	25.6	**40.9**	28.4	29.6	27.5	**39.6**	28.8	29.6	28.0
Edge Pert.	39.2	42.0	39.2	34.2	31.1	34.1	31.1	24.2	33.6	22.9	33.6	24.5	35.1	29.6	35.1	29.5
GDSS [13]	41.3	36.5	41.3	33.4	40.6	40.4	40.6	34.9	40.2	**42.8**	40.2	37.1	35.2	36.8	35.2	32.9
DiGress [26]	38.7	37.4	38.7	37.5	32.5	30.9	32.5	30.0	32.2	31.2	32.2	30.9	31.8	31.9	31.8	31.0
ConGD	43.3	**43.4**	**43.3**	**41.8**	40.7	**43.6**	**40.7**	**39.7**	40.8	42.6	**40.8**	**39.6**	38.2	**39.3**	**38.2**	**37.7**

binary graph with 160 regions of interest (ROIs), comprising 148 cortical and 12 subcortical regions. The dataset consists of 500 subjects with five diagnostic labels, exhibiting severe class imbalance: Control (CN, n=250), Significant Memory Concern (SMC, n=50), Early Mild Cognitive Impairment (EMCI, n=100), Late Mild Cognitive Impairment (LMCI, n=50), and AD (n=50).

Baseline methods. We compared our method with various data augmentation methods, ranging from heuristic edge perturbation to graph generation models (GDSS [13] and DiGress [26]). To assess the general performance improvement from augmentation, we evaluated multiple classification methods, including conventional models such as support vector machines (SVM) and MLP, as well as graph-based models such as GCN [15] and GAT [25]. All results were obtained by averaging results from 10-fold cross-validation. For a fair comparison, the same number of graphs were generated by each augmentation method to balance the class distribution, resulting in 300 samples per class and a total of 1,500 samples.

4.2 Experimental results

Data augmentation using ConGD. We empirically evaluate the effectiveness of the generated data across augmentation methods in Tab. 1. In binary classification, ConGD consistently outperformed all baseline augmentations across every classification methods, achieving average improvement of 4.25%p, 13.43%p, 13.55%p, and 15.18%p in accuracy, precision, recall, and F1-score over no augmentation. For 5-way classification, we observed that standard classification models struggle with class imbalance, exhibiting relatively high accuracy but poor precision and recall, indicating a bias toward dominant labels. Augmentation mitigates this issue with a slight accuracy trade-offs, with ConGD providing



(a) (b)

Fig. 2: t-SNE visualization of the topological embeddings of original and generated brain networks. (a) CN and (b) AD. The samples from ConGD (green) shows the largest overlap with the original data (orange), whereas other methods show distribution shift (red, blue, and purple).

the most substantial averaged gains: 10.8%p in precision, 9.95%p in recall, and 10.35%p in F1-score. These results demonstrate the effectiveness of ConGD in improving class separability via class-conditioned augmentation.

Class-conditioning and Topology-awareness. The quantitative comparison, however, does not explicitly show the improvement of the generated graphs from a topological perspective. Hence, we provide a t-SNE visualization of the topological embeddings of original and generated graphs, obtained using persistent landscape, for different classes (Fig. 2). In both CN and AD, the generated graphs using the edge perturbation method and GDSS [13] formed a distinct cluster, exhibiting poor alignment with the reference graph topology. Compared to DiGress [26], a discrete graph diffusion method, ConGD showed a clear shift of generated graphs toward the reference graphs in the topological space. This highlights the effectiveness of the topology-constraint learning via CIA module and TA regularization.

5 Conclusion

In this work, we introduced ConGD, a topology-constrained conditional graph diffusion framework for Alzheimer stage-specific brain network generation. By integrating conditional factors, i.e., disease label and topological structure, into the generation process, together with additional topological consistency regularization, our method successfully synthesizes brain networks that are highly associated with the conditions as well as resemble the real brain networks in the data distribution. Experimental results on the ADNI dataset demonstrate that ConGD effectively mitigates class imbalance through targeted augmentation, leading to improved downstream classification performance. Our findings highlight the importance of topology-aware learning in brain network synthesis, providing a promising direction for enhancing neuroimaging analyses and expanding the utility of generative models in medical applications.

Acknowledgments. This research was supported by the National Research Foundation of Korea (NRF-2022R1A2C2092336, 50%; RS-2025-02216257, 20%), Institute of Information & communications Technology Planning & Evaluation (RS-2022-II220290, 20%), and AI Graduate Program at POSTECH (RS-2019-II191906, 10%).

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Austin, J., Johnson, D.D., et al.: Structured denoising diffusion models in discrete state-spaces. Advances in Neural Information Processing Systems (2021)
2. Bergmeister, A., Martinkus, K., et al.: Efficient and scalable graph generation through iterative local expansion. In: International Conference on Learning Representations (2024)
3. Bubenik, P.: Statistical topological data analysis using persistence landscapes. Journal of Machine Learning Research (2015)
4. Bullmore, E., Sporns, O.: Complex brain networks: graph theoretical analysis of structural and functional systems. Nature Reviews Neuroscience (2009)
5. Carlsson, G.: Topology and data. Bulletin of the American Mathematical Society (2009)
6. Destrieux, C., Fischl, B., et al.: Automatic parcellation of human cortical gyri and sulci using standard anatomical nomenclature. Neuroimage (2010)
7. Du, S., Lao, Q., et al.: Distilling knowledge from topological representations for pathological complete response prediction. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. pp. 56–65. Springer (2022)
8. Dwivedi, V.P., Bresson, X.: A generalization of transformer networks to graphs. arXiv preprint arXiv:2012.09699 (2020)
9. Edelsbrunner, H., Harer, J.L.: Computational topology: an introduction. American Mathematical Society (2022)
10. Hu, J., Fei, B., et al.: Topology-aware latent diffusion for 3d shape generation. arXiv preprint arXiv:2401.17603 (2024)
11. Hwang, Y., Hwang, S., et al.: Multi-order simplex-based graph neural network for brain network analysis. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. pp. 532–541. Springer (2024)
12. Immonen, J., Souza, A., et al.: Going beyond persistent homology using persistent homology. Advances in neural information processing systems (2023)
13. Jo, J., Lee, S., et al.: Score-based generative modeling of graphs via the system of stochastic differential equations. In: International Conference on Machine Learning (2022)
14. Kerber, M., Morozov, D., et al.: Geometry helps to compare persistence diagrams. ACM J. Exp. Algorithmics (2017)

15. Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks. In: International Conference on Learning Representations (2017)
16. Lacombe, T., Cuturi, M., et al.: Large scale computation of means and clusters for persistence diagrams using optimal transport. Advances in Neural Information Processing Systems (2018)
17. Moor, M., Horn, M., et al.: Topological autoencoders. In: International conference on machine learning. PMLR (2020)
18. Park, J., Hwang, Y., et al.: Convolving directed graph edges via hodge laplacian for brain network analysis. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. pp. 789–799. Springer (2023)
19. Peng, Y., Wang, H., et al.: Phg-net: Persistent homology guided medical image classification. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 7583–7592 (2024)
20. Saggar, M., Sporns, O., et al.: Towards a new approach to reveal dynamical organization of the brain using topological data analysis. Nature Communications (2018)
21. Shehzad, A., Xia, F., et al.: Graph transformers: A survey. arXiv preprint arXiv:2407.09777 (2024)
22. Simonovsky, M., Komodakis, N.: Graphvae: Towards generation of small graphs using variational autoencoders. In: International Conference on Artificial Neural Networks (2018)
23. Sizemore, A.E., Phillips-Cremins, J.E., et al.: The importance of the whole: Topological data analysis for the network neuroscientist. Network Neuroscience (2019)
24. Sporns, O., Tononi, G., et al.: The human connectome: a structural description of the human brain. PLoS computational biology (2005)
25. Veličković, P., Cucurull, G., et al.: Graph attention networks. In: International Conference on Learning Representations (2018)
26. Vignac, C., Krawczuk, I., et al.: Digress: Discrete denoising diffusion for graph generation. In: International Conference on Learning Representations (2023)
27. Wang, F., Liu, H., et al.: Topogan: A topology-aware generative adversarial network. In: European Conference on Computer Vision (ECCV) (2020)
28. Ying, C., Zhao, X., et al.: Boosting graph pooling with persistent homology. In: Advances in Neural Information Processing Systems (2024)
29. You, J., Ying, R., et al.: Graphrnn: Generating realistic graphs with deep auto-regressive models. In: International Conference on Machine Learning (2018)
30. Zhang, Y., Ma, Z., et al.: Topodiff: Improving protein backbone generation with topology-aware latent encoding. bioRxiv (2023)